

The Fault Management System Using Amazon EC2: Web Application Deployment on Cloud

K Raja Sekhar, M.Tech (PhD) Sarat Chalamalsetty Prahasith Katuri Varun Jajam

Department of Computer Science
KL University.AP

Abstract-- Cloud computing is a technology that uses the internet and central remote servers to maintain data and applications. Fault Management System is a key part in Network Management Architecture consists of functions such as detect, isolate, determine the cause and correct the malfunctions in network. The entire application is deployed on cloud. Cloud is used for its per-use pricing and elastic scalability, and a significant advantage over the typical acquisition and deployment of equipment in traditional servers, which were previously used. Amazon Web Services is one of the cloud providers that provides virtual server instances managed through a web service API with unique IP addresses and host names assigned dynamically and blocks of storage on demand.

Keywords—Amazon EC2; Fault Management System, EC2 command Line tool, EC2 QRC

I. INTRODUCTION

Cloud computing is a new paradigm for IT that displaces the current dominant paradigm, enterprise computing. The advantages of enterprise computing over cloud computing like customization, load balancing, Content Delivery, Failover, Scalability, Tooling and on demand service drives us to deploy our web application (fault management) onto the cloud.

A. Fault management

Fault management is the set of functions that discover, isolate, and correct malfunctions in a telecommunications network, compensate for environmental changes, and include maintaining and examining error logs, accepting and acting on error detection notifications, carrying out sequences of diagnostics tests, correcting faults, reporting error conditions, and localizing and tracing faults by examining and manipulating database information

In this project once error is reported, first step is to detect the fault and make a report of it. Based on this report we will isolate the fault i.e. to find the exact location of the fault and to determine the cause of the fault. Once isolated the fault need to be corrected and appropriate action need to be taken and report is send to administrator. The administrator will maintain error logs and keep track of the person who works on the fault

B. Amazon EC2

Survey

Cloud computing is a prevalent term for anything that involves delivering hosted services over the Internet. These services are broadly divided into three categories: Infrastructure-as-a-Service (IaaS), Platform-as-a-Service (PaaS), and Software-as-a-Service (SaaS). There are many Cloud providers and among them Amazon Elastic Compute Cloud (Amazon EC2) is a web service that provides resizable compute capacity in the cloud. It is designed to make web-scale computing easier for developers.

Amazon EC2's simple web service interface allows us to obtain and configure capacity with minimal friction. It provides us with complete control of your computing resources and lets us run on Amazon's proven computing environment. Amazon EC2 reduces the time required to obtain and boot new server instances to minutes, allowing you to quickly scale capacity, both up and down, as our computing requirements change. Amazon EC2 changes the economics of computing by allowing you to pay only for capacity that you actually use. Amazon EC2 provides developers the tools to build failure resilient applications and isolate themselves from common failure scenarios. Amazon EC2 presents a true virtual computing environment, allowing us to use web service interfaces to launch instances with a variety of operating systems, load

them with your custom application environment, manage your network's access permissions, and run your image using as many or few systems as you desire.

Usage

To use Amazon EC2, we simply create an Amazon Machine Image (AMI) containing your applications, libraries, data and associated configuration settings Or use pre-configured, template images to get up and running immediately. Upload the AMI into Amazon S3. Amazon EC2 provides tools that make storing the AMI simple. Amazon S3 provides a safe, reliable and fast repository to store your images. Use Amazon EC2 web service to configure security and network access. Choose which instance type(s) and operating system you want, then start, terminate, and monitor as many instances of our AMI as needed, using the web service APIs or the variety of management tools provided. Determine whether we want to run in multiple locations, utilize static IP endpoints, or attach persistent block storage to your instances. Pay only for the resources that you actually consume, like instance-hours or data transfer.

II. DESIGN GOALS AND OBJECTIVES

A. Design assumptions

The Web App is intended to avoid server failure, virus attack and decrease the infrastructure costs, maintenance costs occurred to maintain the application

In designing a storage administration system to meet our goals, we have been guided by modules offering both challenges and opportunities

The modules used here are:

Module-1: Designed to provide authentication to individuals who wish to get connected. The registration details are stored in the database and whenever the user logs in, the user credentials are retrieved to check whether the user is an authorized user or not. The front end is designed using Eclipse.

Module-2: The error is reported by the customer, a ticket is generated for the fault. Manager views the faults, escalates the faults to available analyzer or technicians. He monitors the fault management system. He maintains error log and keep track of the problem and person who is working on it.

Module-3: Analyzer and Technicians solve the problems allocated by the manager depending on the location from where the error encountered and generate reports after rectifying them.

Module-4: Administrator maintains error log and keep track of the problem and person who is working on it.

Module-5: In this module we will deploy entire application in Amazon Elastic Compute Cloud (EC2) of Amazon Web Service.

B. Objectives:

- Manage the application on cloud
- Creating a User Interface for Fault management system on cloud
- The process involves Registration, Authentication and operator will get the complaint.
- This application or the UI is converted into Amazon Machine Image(AMI)
- Deploying the AMI on the Amazon Cloud
- The benefits we gain are, there is no extra maintenance cost incurred. The application will be maintained by the cloud. No need to of extra infrastructure cost. Avoid virus attack No server failure.

III. ARCHITECTURE

When a fault or event occurs, a network component will often send a notification to the network operator using a protocol. Fault Management System helps in increasing network availability, reduce network downtime and restore network failure quickly. Based on this report we will isolate the fault i.e. to find the exact location of the fault and to determine the cause of the fault. Once isolated the fault need to be corrected and appropriate action need to be taken and report is send to administrator. Finally the entire application is deployed on cloud.

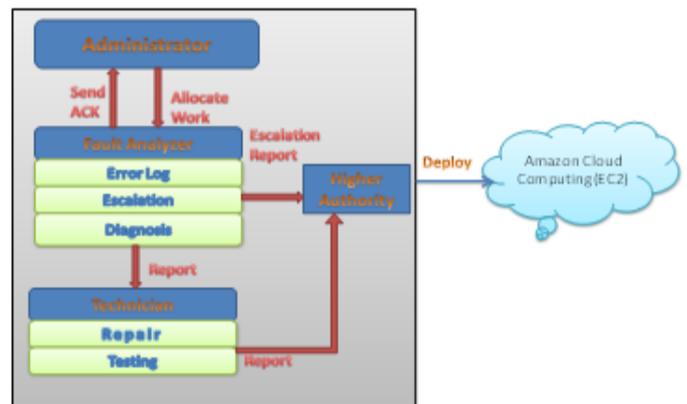


Fig. 1. Cloud Based Fault Management System

a. *Administrator*-The administrator will maintain the error logs, the users of the system, the employee registration, the work that is allocated to the users and the protocols that are followed for eradicating the faults.

b. *Fault Analyzer*-The fault analyzer checks the fault, sends the acknowledgement, updates the error log, escalates the fault and diagnoses the fault. After thorough diagnosis it generates a report. The report is then sent to the Technician.

c. *Technician*-The technician checks the report and starts to work on that fault. He repairs the fault tests it for further errors and then generates a report and it is sent to the higher authority for confirmation.

IV. DEPLOYMENT

The default small instance will be provided with its 1.7GB of RAM and 160GB disk. 10 GB of Amazon Elastic Block Storage, plus 1 million I/Os, 5 GB of Amazon S3 standard storage, 20,000 Get Requests, and 2,000 Put Requests, 15 GB of bandwidth out aggregated across all AWS services will be provided on free usage tier. After downloading command tool, use

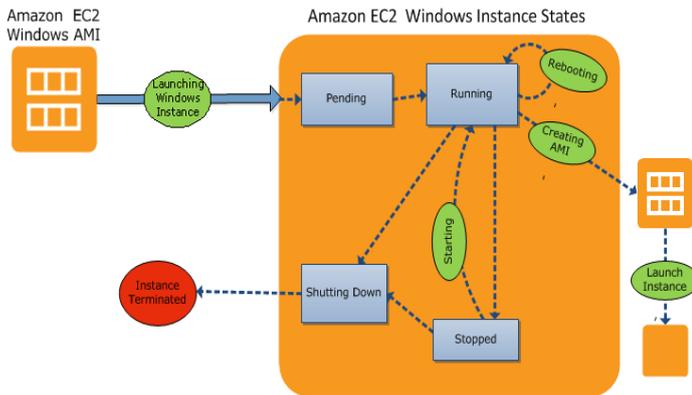


Fig. 2 Amazon EC2 Windows Instance

Command Line connectivity: (EC2 command line tool)

A. Step by Step connection:

1. Set the JAVA_HOME environment variable:

Install Java 1.5, Set JAVA_HOME to the full path of the directory that contains a subdirectory named bin that in turn contains the Java executable.

Example, if your Java executable is in C:\jdk\bin, set JAVA_HOME to C:\jdk.

On the machine from which you will connect to Amazon Web Services, go to Environment Variables.

Under System variables

Variable name: JAVA_HOME

Variable value: Path to your Java version.

Example: "C:\Program Files(x86)\Java\jre7".

Add your Java directory to your path before other versions of Java. In System variables, select Path, and then click Edit. In Variable values, before any other versions of Java add %JAVA_HOME%\bin;

Verify your JAVA_HOME setting with the command %JAVA_HOME%\bin\java -version.

2. Set the EC2_HOME Environment Variable:

Set EC2_HOME to the path of the directory into which you unzipped the command line tool. This directory is named ec2-w.x.y.z (w, x, y, and z are

version/release numbers) and contains sub-directories named bin and lib. Go to environmental variables.

System variables:

Variable name: EC2_HOME.

Variable value: Path to the directory where you installed the command line tools.

Example, "C:\AWS\EC2\ec2-1.0.12.0".

Add the tool's bin directory to your system PATH. In the same System variables, select Path, and then click Edit.

Variable values: add; %EC2_HOME%\bin.

3. Set the EC2_PRIVATE_KEY and EC2_CERT Environment Variables

To set up security credentials for your command line tool

a. If you want to create a new certificate and private key, log in to the AWS security credentials website. Go to the Access Credentials section and select the X.509 Certificates tab. Click Create a new Certificate, The X509 Certificate Created page appears. Store your Private Key file in a secure location. Select Download X.509 Certificate and save it to a convenient location on your computer.

b. Set the EC2_CERT environment variable to the fully qualified path of an existing X.509

Go to environmental variables

System variables:

Variable name: EC2_CERT.

Variable value: Path to the directory where you saved your certificate.

Example:

C:\aws\certs\cert\pk-B2QQ72ELI6RDHOOYX6RK6ZPAW2SXTFUX.pem.

c. Set the EC2_PRIVATE_KEY environment variable to the fully qualified path of the private key that is associated with the X.509 certificate that EC2_CERT now points to.

Go to Environment Variables,

Under System variables

Variable name: EC2_PRIVATE_KEY.

Variable value: Path to the directory where you saved your private key. Example,

C:\aws\certs\key\pk-pk-LJFUGMXPQD6VAHJBBQFCQB7KHP4XPX.pem

4. Set the Region

a.To view available regions go to Regions and Endpoints in the Amazon Web Services General Reference.

b. If you want to change the service endpoint, set the EC2_URL environment variable.

The following example sets EC2_URL to the APAC(Singapore) Region permanently.

```
C:\> setx EC2_URL
https://ec2.apac1.amazonaws.com
```

There are four Amazon EC2 regions:
 US (N. Virginia), US (N. California), EU (Ireland), APAC (Singapore).

B. Task List for Creating an Amazon EBS-backed AMI

1. Start by launching an instance of an Amazon EBS-backed AMI that is similar to the AMI we want to create. For example, we might take a public AMI that uses the operating system we want to use for your AMI. The instance must be from an Amazon EBS-backed AMI; we can't start with an instance of an Amazon EC2 instance store-backed AMI.

2. When the instance is running, customize it as you want. For example, you could attach additional Amazon EBS volumes, or load applications or data on to the instance.

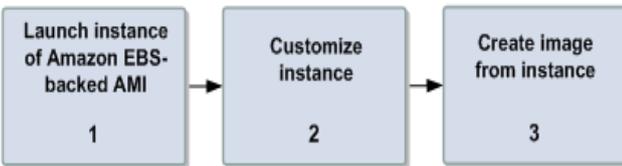


Fig. 3 Creation of Amazon EC2 backed Windows AMI

3. To create an Amazon EBS-backed AMI

1. Enter the following command to create an image:

```
C:\> ec2-create-image -n your_image_name
instance_id
```

Example:

```
C:\> ec2-create-image -n "My AMI" i-eb977f82
```

Amazon EC2 creates an image and returns an AMI ID.

```
IMAGE ami-8675309
```

2. If you want to check whether the AMI is ready, enter the following command:

```
C:\> ec2-describe-images -o self
```

After Creation of AMI, by performing Installation of Firefox Plugin, Launching of Instance, Testing Web

Server, Adding SSH to Server, we have just deployed a Fault management application on an Amazon EC2 instance.

Instance Tools	
ec2-run-instances (ec2run) Launches one or more instances of the specified AMI. ec2-run-instances ami_id [-n instance_count] [-g group [-g group ...]] [-k keypair] [-d user_data -f user_data_file] [-t instance_type] [-z availability_zone] [--kernel kernel_id] [--ramdisk ramdisk_id] [--block-device-mapping] [--monitor] [--disable-api-termination] [--instance-initiated-shutdown-behavior behavior] [--placement-group placement_group] [--tenancy tenancy] [-s subnet] [--private-ip-address ip_address]	ec2-terminate-instances (ec2kill) Terminates the specified instance. ec2-terminate-instances instance_id [instance_id ...]
ec2-describe-instances (ec2din) Lists the specified instances. If no instance is specified, all your instances are listed. ec2-describe-instances [instance_id ...] [--filter name=value] ...	ec2-start-instances (ec2start) Starts a stopped instance (applies only to Amazon EBS-backed instances). ec2-start-instances instance_id [instance_id ...]
	ec2-describe-instance-attribute (ec2dinatt) Describes an attribute for the specified instance. ec2-describe-instance-attribute instance_id [-b block-device-mapping --disable-api-termination --instance-initiated-shutdown-behavior --instance-type --kernel --ramdisk --root-device-name --user-data]
Amazon EBS Tools	
ec2-create-volume (ec2addvol) Creates an Amazon EBS volume from a snapshot or an empty volume in the size you specify. ec2-create-volume -s availability_zone [-s size --snapshot snapshot]	ec2-create-snapshot (ec2addsnap) Creates a snapshot of an Amazon EBS volume and stores it in Amazon S3. ec2-create-snapshot volume_id
ec2-describe-volumes (ec2dvol) Lists the specified Amazon EBS volumes. If no volume is specified, all your volumes are listed. ec2-describe-volumes [volume_id ...] [--filter name=value] ...	ec2-describe-snapshots (ec2dsnap) Lists the specified snapshots. If no snapshot is specified, all your snapshots are listed. ec2-describe-snapshots [snapshot_id ...] [--filter name=value] ...
ec2-delete-volume (ec2delvol) Deletes the specified Amazon EBS volume. ec2-delete-volume volume_id	ec2-delete-snapshot (ec2delsnap) Deletes the specified Amazon EBS snapshot. ec2-delete-snapshot snapshot_id
ec2-attach-volume (ec2attvol) Attaches an Amazon EBS volume to a running instance and exposes it as the specified device. The volume and instance must be in the same Availability Zone. ec2-attach-volume volume_id -i instance_id [-d device]	ec2-detach-volume (ec2detvol) Detaches an Amazon EBS volume from an instance. ec2-detach-volume volume_id [-i instance_id [-d device]] [--force]

Fig. 4 Amazon EC2 Quick Reference Card

The figure represents some of the commands that are needed for the creation of the instances, volumes, images and allocation of addresses.

V. CONCLUSION AND FUTURE WORK

Service Developers want the Service Providers to ensure or provide the capability to dynamically allocate and manage resources in response to changing demand patterns. It is now possible to provide dynamic visibility and control of services management to meet the rapidly growing demand for Amazon Web Service.

This software is developed using Eclipse, WebSphere, DB2, and AmazonEC2. The goals achieved by the deployment of our software in cloud are Instant Access, Improved Productivity, Optimum Utilization of resources, Efficient management of records, Simplification of the operations, Less processing time and required information, Portable and flexible for further enhancements.

Intensive care has been taken by the development team to meet all the requirements of the client. Each module has undergone stringent testing procedures and the integration testing activity has been performed by each module.

After completion of execution, we confirm that the Product has reached the requirements. Finally we conclude that the project stands up in maintaining its sole motive of betterment to the society and work for a social cause. Our current application deployment makes us to deploy any application into cloud as a simple task. In our future plan, we will try to build various web applications which direct to handle various QoS requirements.

ACKNOWLEDGMENT

This work is supported by KL University, Andhra Pradesh. We thank Varun.Budda for his useful suggestions while creating the AMI. We wish to avail ourselves of this opportunity, express a sense of gratitude to our beloved parents for their manual support, strength, help and for everything .Finally, we give thanks to the all the people who played a part in use.

REFERENCES

- [1] Michael Miller, "Cloud Computing: Web-Based Applications That Change the Way You Work and Collaborate Online", Que; 1 edition (August 21, 2008)
- [2] Toby Velte, "Cloud Computing, A Practical Approach"
- [3] George Reese, "Cloud Application Architectures: Building Applications and Infrastructure in the Cloud" (Theory in Practice (O'Reilly)), O'Reilly Media; 1 edition (April 10, 2009)
- [4] Judith Hurwitz , Robin Bloor, Marcia Kaufman, Fern Halper, "Cloud Computing For Dummies" ,For Dummies; 1 edition (November 16, 2009)
- [5] Dan Kusnetzky, "ZDNet, Nasuni Cloud Storage Gateway", June 1, 2010
- [6] Branter, Florescu, Graf, Kossmann, Kraska, "Building a database on S3", 2008.
- [7] Sobel, Subramanyam, Sucharitakul, Nguyen, "Cloudstone: Multi-Platform, Multi-Language Benchmark and Measurement Tools for Web 2.0". , et al. 2008
- [8] Broberg, Buyya, Tari, Creating a 'Cloud Storage' Mashup for HighPerformance, Low Cost Content Delivery. 2008
- [9] Buyya, Shin Yeo, Venugopal, Broberg, Brandic, Cloud Computing and Emerging IT Platforms:Vision, Hype, and Reality for Delivering Computing as the 5th Utility, 2008.
- [10] Barham, Dragovic, Fraser, Hand, Harris, "Xen and the art of virtualization", 2003.
- [11] Rich Miller (2008-05-02). "IBM, Google Team on an Enterprise Cloud". DataCenterKnowledge.com. <http://www.datacenterknowledge.com/archives/2008/05/02/ibm-google-team-on-an-enterprise-cloud/>.
- [12] Sean Rhea, Chris Wells, Patrick Eaton, Dennis Geels, Ben Zhao, Hakim Weatherspoon, and John Kubiatowicz, Maintenance-Free Global Data Storage. IEEE Internet Computing, Vol 5, No 5, September/October 2001 , pp 40-49.
- [13] Cloud Confusion Amongst IT Professionals. VersionOne (June 6, 2011).
- [14] Amazon Web Services. "Innovation. Powered by Amazon Web Services". <http://aws.amazon.com>